# CS 255 Business Requirements Document Template

## System Components and Design

### Purpose

* The purpose of this application is to fill a void in the DMV training. The application will aid in the training of users on how to drive by the law. The client’s are Liam and Ian from the company DriverPass.

### System Background

* DriverPass is to have an application for a company to reserve, manage, track training sessions. It also allows for users to study and take practice test to pass the driving test. They will need roles in the system to manage. They will also need to setup trainees with the trainers and manage where and when to pick up and drop off trainees.

### Objectives and Goals

* The system should be able to train user on how to abide by driving laws and pass the DMV drivers test. The measurable task are the outline of the plan, the interface, scheduling appointments, user roles, practice test and study guides.

## Requirements

### Nonfunctional Requirements

#### Performance Requirements

* The program should be an application ran program that accesses a server that updates the app with new information. This would allow for the user to see their information without the need of data. It would also reduce the load of the server since the interactions would only happen when the app refreshes the data and a connection test to the server. The system should be updated every 3 months or unless there is a vulnerability that needs an immediate patch.

#### Platform Constraints

* The platform should be on windows since it is the most versatile and relatively low cost to start up. It also allows for a large amount of driver software to go across other operating systems. There would need to be a database that is based off the current DMV testing information. There would also need to be a database that stores the user’s information, car statuses, and employee schedules for appointments.

#### Accuracy and Precision

* The username will be the way to distinguish users. This creates a single variable that once verified can be used to interact with the program. User input will not need to be case-sensitive except for the password. The password should be case-sensitive for security purposes. The application should always be taking a log of the system. Any error should be presented to the admin and then create a course of action to fix or improve.

#### Adaptability

* An admin should be able to add and remove a user without changing code. Modifying the user should only be done by the user to ensure that everyone is kept honest. The system will create a process to add and remove user from the administrator account. This will keep the databases up to date and clean from extra data. An IT admin needs to have an administrator account to have the ability to update, interact, and test the system to it’s full capability.

#### Security

* The system must be secure since the school is paid for. Users banking information, name and address are also stored in the program. To login the recommended type would be too have a username and password. Along with having that some type of MFA to reduce the chances of unwanted access. If an account is attacked it should be moved into a quarantine section along with requiring the user to create a new username and password. If the user forgets their password a passcode should be sent to the accounts registered email or a phone call confirmation. After the confirmation call or correct passcode a new browser window should open that allows the user to reset their password.

### Functional Requirements

* The system will have an administrator user role to manage, view and test functionality and security.
* The system will have a content manager user role to create classes, manage resources, produce lesson plans, etc.
* The system will have a user role to have an account that can verify the user, interact with classes, and create appointments.
* The system will have a notification module to alert admins of possible appointment conflicts. It must also send notices to the content manager for updates in the user interactions with the system.
* The system shall have a class module that stores the content and must be able to take user interaction and relay that information to the content manager to grade and give feedback.
* The system shall have a student portal that presents a course to the user an provide a overview of all the classes that the user is enrolled in.

### User Interface

* The interface needs have a simple interface that allows for easy navigation of the program. This will help with ease of use. Based off the account that is logged in the program will have more and more options. The admin account will have the most access to manage and test the program. The user account will have the option to make an appointment, partake in classes and study the course information. The mobile and browser interface should be as similar as possible to make the transition between the 2 as smooth as possible.

### Assumptions

* Making online classes dynamic will provide flexibility because of the availability of users. Since users needs to always have access to the class the system must be dynamic. Also, users will be using different types of devices to access the application.
* There must be multiple options of the classes and study guides to accommodate for different types of learners. This will create a more welcoming class and retain customers.

### Limitations

*Any system you build will naturally have limitations. What limitations do you see in your system design? What limitations do you have as far as resources, time, budget, or technology?*

* The focus type of the program must be broad since the program cannot be adjusted to each individual user. This has drawbacks because some users will not find the program helpful while other users live off the program.
* Since it is a web-based application there are reduced informal learning possibilities. Outside of the one-on-one driving lessons all other classes are held online.

### Gantt Chart

Timeline

Description automatically generated with medium confidence